

---

# RETOUR SPEC

---

## Erreurs :

Les points de dégâts des cartes d'un champ de bataille ne sont pas réinitialiser au début de son tour.

---

La fonction `reinitDegat(joueur)` est ajoutée au fichier `joueur`. Elle modifie les points de dégâts subis par toutes les cartes à la fin du tour

Lorsque une carte est capturée dans le royaume on doit aussi remettre les dégâts à 0 en prévision d'un futur déploiement.

---

Pas un soucis dans le cas de la capture car ses points de dégâts sont déjà à 0 et on ne lui inflige pas pour faire la capture, on ne fait que déplacer la carte.

Il n'y a pas de fonction `créerCarte()`

---

Problème résolu dans le fichier `carte.py`

Dans la mise en réserve si il y a 5 cartes demander si on veut placer la carte sélectionner dans le royaume.

---

Les règles indiquent que, lors de la phase d'action « Mettre en reserve » si il y a 5 cartes dans la réserve alors on ne peut rien faire, on ne pose pas la carte en réserve ni dans le royaume ! On ne peut juste pas la poser.

Nous avons mis en place une fonction `estPleineReserve(reserve)` qui indique si la réserve est pleine ou non. (Elle se trouve dans `reserve.py`)

Dans certaines fonctions comme `extraireCarteMain` on nous demande d'entrer un indice `i` ce qui suppose une structure de données.

---

Ceci ne suppose absolument pas une structure de données ! L'indice correspond à ce que l'utilisateur a vu renvoyé par `DescriptionMain`. Rien ne vous force à utiliser un tableau une liste ou quoi que ce soit !

Dans la fonction `creerMain`, préciser que l'indice doit être entre 1 et 2

---

Modifié dans la spécification fonctionnelle de la fonction

## Pour améliorer :

**Vous pouvez empêcher le joueur d'attaquer si aucune carte adverses n'est à portée.**

---

Pas besoin, c'est à l'utilisateur de faire attention. Dans le déroulement du jeu, si le joueur choisi une carte attaquante sans que la cible soit dans sa portée, alors l'attaque échoué. Pas de répit !

**Si le roi est capturé le programme demande encore si le joueur veut continuer d'attaquer.**

---

Modifié. Avant de demandé si le joueur veut attaquer la même cible ou une autre, on vérifie que la partie ne soit pas déjà finie. Rappelons que avant de lui demander s'il veut attaquer la même cible, on vérifie que la cible soit toujours en combat. Rappelons aussi qu'avant de lui demander s'il veut attaquer une autre cible, on vérifie qu'il puisse encore attaquer (touteHorizontale())